

## OBJEKTI. ENKAPSULACIJA.

**Uputa za rješavanje:** Napraviti Solution *ProjektiVjezba*, a u njemu dodati novi projekt za svaki od zadataka. U svim programima treba predvidjeti i osigurati se od mogućih pogrešaka uz odgovarajuću poruku.



### 1. Proizvod

Napravi klasu proizvod koja ima svojstva:

- Naziv (string)
- Cijena (double)
- Marža (double)
- Porez (double)

i metodu:

- $MPC = Cijena + Marža + Porez$

U glavnim programu kreiraj objekt klase Proizvod, traži od korisnika unos polja i zatim ispiši MPC. Program osiguraj od grešaka.



### 2. Automobil

Napravi klasu Automobil koja ima svojstva:

- Marka (string)
- KS (double)
- OsnovnaCijena (double)

i metode:

- IznosPoreza (double)
  - 5% ako je  $KS \leq 50$
  - 10 % ako je  $50 < KS \leq 150$
  - 15% ako je  $KS > 150$
- Ukupna cijena = OsnovnaCijena + IznosPoreza



### 3. Zaposlenik

Napravi klasu Zaposlenik koja ima svojstva:

- Ime
- Prezime
- OIB
- BrojBodova
- VrijednostBoda
- Porez (Read Only svojstvo)
  - 5 % ako je  $NetoPlaca \leq 3000$
  - 10 % ako je  $3000 < NetoPlaca \leq 6000$
  - 20 % ako je  $NetoPlaca > 6000$

Metode:

- $NetoPlaca = BrojBodova * VrijednostBoda$
- $BrutoPlaca = NetoPlaca + Porez$

Konstrukture:

- Defaultni

- Konstruktor s parametrima: ime i prezime
- Konstruktor s parametrima: ime prezime i OIB



#### 4. Kalkulator

Napravi klasu sa static metodama:

- Zbroj (double broj1, double broj2)
- kub (double broj)
- UdaljenostTocaka (double x1, double x2, double y1, double y2)
- CelzijFahrenheit (int stupnjevi)

Napomena: Klasa se ne smije moći instancirati



#### 5. StaticsInstancom

Napravi klasu koja ima

- static funkciju koja vraća je li broj prost ili složen
- metodu na instanci koja ispituje je li broj savršen (jednak zbroju svih svojih djelitelja osim njega samoga)



#### 6. GDPR

Napravi klasu Student (u odvojenoj datoteci) sa svojstvima:

- Ime
- Prezime
- GodinaRodjenja (kod unosa se očekuje bez točke na kraju)
- MjestoStudiranja
- OIB

I metodama:

- Inicijali – vraća inicijale studenta
- Starost – vraća starost studenta u godinama
- ProvjeriOIB – provjerava je li uneseni OIB ispravan (ima 11 znamenaka)

U glavnom programu napravi listu objekata tipa Student i omogući unos novog studenta sve dok se za ime ne upiše „kraj“. Nakon unosa za svakog studenta u listi ispiši inicijale, starost i je li OIB ispravan. (Dodatnu ispravnost OIB-a možeš provjeriti prema uputama na web stranici:

<https://regos.hr/app/uploads/2018/07/KONTROLA-OIB-a.pdf> )

Na kraju ispiši sve studente koji studiraju u Osijeku i stariji su od 24 godine.